PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-020501
(43)Date of publication of application : 21.01.2000

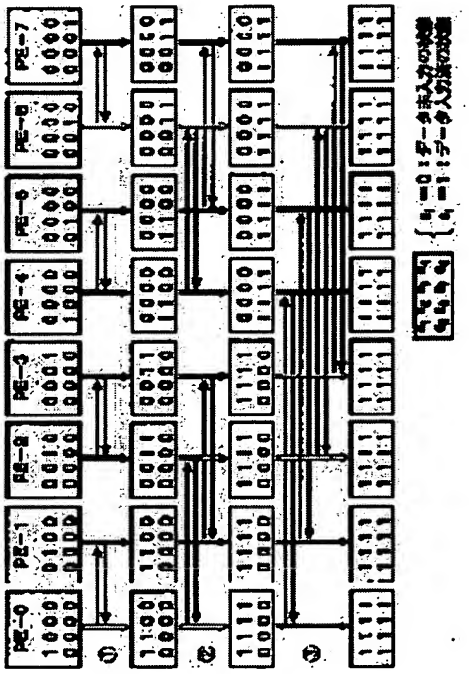
(51)Int.CI. G06F 17/10
G06F 15/16

(21)Application number : 10-188840
(22)Date of filing : 03.07.1998

(71)Applicant : TOSHIBA CORP
(72)Inventor : UEMATSU MIKIO

(54) PARALLEL COMPUTER SYSTEM AND COMMUNICATION METHOD BETWEEN ARITHMETIC PROCESSING UNITS

(57)Abstract:
PROBLEM TO BE SOLVED: To efficiently collect data distributively processed by respective arithmetic processing units(APU) in a parallel computer.
SOLUTION: In the case of collecting data arrays divided into 2n small arrays and distributed/computed by respective APUs as one array in the parallel computer system provided with 2n APUs to which identification(ID) numbers 0, 1,..., 2n, −1 are given, individual storage means and communication means, a number N' obtained by inverting the value of 2i−th digit of an ID number N expressed by binary notation is allowed to correspond to the ID number N and operation (i) for mutually transmitting/receiving the arithmetic processing results of data arrays between the APU of the ID number N and the APU of the ID number N' is successively executed from i=0 to i=n−1. When j>0, arithmetic processing results obtained up to operation (j−1) are transmitted/ received between the APUs of the ID numbers N, N' in addition to the arithmetic processing results obtained by respective APUs at the time of operation (i).



LEGAL STATUS

[Date of request for examination] 01.06.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2000－20501
（Ｐ２０００－２０５０１Ａ）
(43)公開日　平成12年１月21日(2000.1.21)

(51)Int.Cl.[7]	識別記号	ＦＩ		テーマコード（参考）
G06F 17/10		G06F 15/31	Z	5B045
15/16	390	15/16	390 Z	5B056

審査請求　未請求　請求項の数７　ＯＬ　（全17頁）

(21)出願番号　特願平10－188840

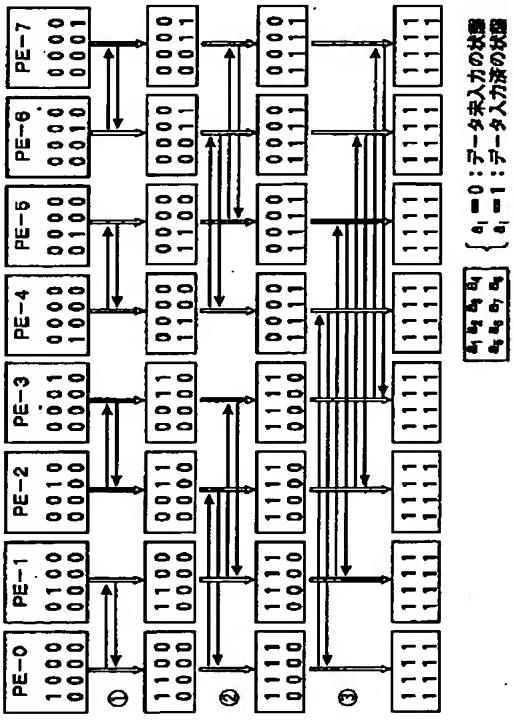
(22)出願日　平成10年７月３日(1998.7.3)

(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地
(72)発明者　上松　幹夫
神奈川県横浜市磯子区新杉田町８番地　株式会社東芝横浜事業所内
(74)代理人　100083161
弁理士　外川　英明

Ｆターム(参考)　5B045 AA07 BB02 BB28 BB47 GG02
GG12
5B056 AA04 BB45 DD14 FF05

(54)【発明の名称】並列計算機システム及びその演算処理装置間の通信方法

(57)【要約】
【課題】　並列計算機の各演算処理装置で分散処理されたデータを効率的に集結する。
【解決手段】　識別番号０，１，…，2^n-1が付与された2^n台の演算処理装置と個別記憶装置及び通信手段を備えた並列計算機システムで，2^n個の小配列に分割して各演算処理装置に分配／演算処理されたデータ配列を１つの配列に集結する際に，識別番号Ｎに対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を対応させ，識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間でデータ配列の演算処理結果を相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行う。この際，ｊ＞０なるｊに対しては，操作ｉの際に，識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信する。



1

【特許請求の範囲】

【請求項１】　固有の識別子を有する少なくとも2^n台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，2^n個の小配列に分割して2^n台の演算処理装置に分配され各演算処理装置で演算処理されたデータ配列を再び１つの配列に集結する際に，2^n台の演算処理装置に識別番号０，１，…，2^n-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対しては，操作ｊの際に，識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することにより2^n台の演算処理装置間でｎ回の操作でデータ配列を集結させることを特徴とする並列計算機システム。

【請求項２】　固有の識別子を有する（2^n+k）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，（2^n+k）個の小配列に分割して（2^n+k）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，前記（2^n+k）台の演算処理装置に個別記憶手段及び通信手段を備えた（2^n-k）台の演算処理装置を加えた2^{n+1}台からなる演算処理装置群を形成し，この演算処理装置群を構成する2^{n+1}台の演算処理装置に識別番号０，１，…，$2^{n+1}-1$を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｍまで順次行い，ｊ＞０なるｊに対しては，操作ｊの際に，$N \leqq 2^n+k$なる識別番号Ｎの演算処理装置からはその演算処理装置の演算処理結果及び操作（ｊ－１）までで得られた演算処理結果を送信し，$N > 2^n+k$なる識別番号Ｎの演算処理装置からは操作（ｊ－１）までで得られた演算処理結果を送信することにより（2^n+k）台の演算処理装置において（ｍ＋１）回の操作でデータ配列を集結させることを特徴とする並列計算機システム。

【請求項３】　固有の識別子を有する（2^n+k）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機

2

システムにおいて，（2^n+k）個の小配列に分割して（2^n+k）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，この（2^n+k）個のデータ配列に（2^n-k）個の空の小配列を追加することで前記データ配列を小配列2^{n+1}個分の配列に拡張し，前記（2^n+k）台の演算処理装置に，個別記憶手段及び通信手段を備えた（2^n-k）台の演算処理装置を加えた2^{n+1}台からなる演算処理装置群を形成し，この演算処理装置群を構成する2^{n+1}台の演算処理装置に識別番号０，１，…，$2^{n+1}-1$を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｍまで順次行い，ｊ＞０なるｊに対して，操作ｊの際に，識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することにより（2^n+k）台の演算処理装置において（ｍ＋１）回の操作でデータ配列を集結させることを特徴とする並列計算機システム。

【請求項４】　ｎ＞ｍなるｎ，ｍについて，固有の識別子を有する（2^n+2^m）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，（2^n+2^m）個の小配列に分割して（2^n+2^m）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，前記（2^n+2^m）台の演算処理装置を2^n台からなるグループG_1と2^m台からなるグループG_2に分割し，また前記データ配列を初めの2^n個の小配列からなる配列A_1とその後の2^m個の小配列からなる配列A_2の２つに分割し，この配列A_1，A_2をそれぞれグループG_1，G_2と対応づけて分配，演算処理を行い，グループG_1の2^n台の演算処理装置に識別番号０，１，…，2^n-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_1内で配列A_1を集結させる第１の工程と，グループG_2の2^m台の演算処理装置に識別番号０，１，…，2^m-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識

別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_1内で配列A_2を集結させる第２の工程と，グループG_1からグループG_2の各演算処理装置に配列A_1を，グループG_2からグループG_1の各演算処理装置に配列A_2を送信する第３の工程とを有し，第１の工程と第２の工程を並列に実行した後に第３の工程を行なうことにより（$2^{n_1}+2^{n_2}$）台の演算処理装置においてデータ配列を集結させることを特徴とする並列計算機システム。

【請求項５】　固有の識別子を有する複数の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備えた並列計算機システムにおいて，

【数１】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

個の小配列（但し，$n_1>n_2>n_3>\cdots>n_k\geqq 0$）に分割して

【数２】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，これらの演算処理装置のうち

【数３】

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

台をそれぞれグループG_1，G_2，…，G_kとしてｋ個のグループに分割するとともに，前記小配列のうち

【数４】

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

個の小配列をそれぞれ配列A_1，A_2，…，A_kとしてｋ個の配列に分割し，このｋ個の配列とｋ個のグループG_1，G_2，…，G_kとを１対１に対応づけて分配・演算処理を行い，$1\leqq p\leqq k$なる各ｐに対し，グループG_pの（2^{n_p}乗）台の演算処理装置に識別番号０，１，…を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_p内の演算処理装置でデータ配列A_pを集結させるグループ内工程ｐを実行し，グループ内工程（ｋ－１）が終了した後，グループG_kの演算処理装置から配列A_kの演算結果をグループG_{k-1}の演算処理装置に送信するグループ間工程ｋを実行し，次に，グループG_pの各演算処理装置に集結された配列A_pの演算結果を，グループG_pの演算処理装置からｑ＞ｐなる全てのｑに対しグループG_qに属する各演算処理装置に送信するとともに，グループG_qの演算処理装置から，グループG_p自身の演算結果である配列A_p及びグループG_{p+1}の演算処理装置から受信した配列A_{p-1}，…，A_kの演算結果をグループG_{p-1}の演算処理装置に送信するグループ間工程ｐを，ｐ＝ｋ－１からｐ＝２までｐに関して降順に実行することにより，

【数５】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

台の演算処理装置においてデータ配列を集結させることを特徴とする並列計算機システム。

【請求項６】　請求項４または５記載の並列計算機システムを用いて演算処理装置のグループ間でのデータ交換を行う場合，ｐ＞ｑなるｐ，ｑについて，2^p台の演算処理装置からなるグループG_Aで集結され共有されているデータ配列Ａと，2^q台の演算処理装置からなるグループG_Bで集結され共有されているデータ配列Ｂとを，グループG_A，G_B間で相互に送受信する際に，グループG_Aのなかから選択される2^q台の演算処理装置をグループG_Bの各演算処理装置と１対１に対応させてグループG_Bの各演算処理装置にデータ配列Ａを送信する操作を並列に実施するとともに，グループG_Aを，それぞれが2^{p-q}台の演算処理装置からなる小グループα_1，α_2，…，α_r（$r=2^q$）に分割して，各々の小グループとグループG_Bのｒ台の各演算処理装置とを１対１に対応させ，小グループα_iのなかから選択される１台の演算処理装置に対して，小グループα_iに対応するグループG_Bの演算処理装置からデータ配列Ｂを送信した後，小グループα_iの演算処理装置間でデータ配列Ｂを送受信する操作ｉを，$1\leqq i\leqq r$なるｉに関して並列に実行するすることにより，2^p台の演算処理装置と2^q台の演算処理装置にデータ配列Ａとデータ配列Ｂを共有させることを特徴とする並列計算機システム。

【請求項７】　請求項１乃至６のいずれか記載の並列計算機システムを用いて２台の演算処理装置間でデータを交換する工程は，演算処理装置の識別番号の大きい方から小さい方にデータを送る第１の送信工程と，演算処理装置の識別番号の小さい方から大きい方にデータを送る第２の送信工程とからなり，この第１の送信工程と第２の送信工程のうちから選択される１工程を先に行った後，続いて他の１工程を行うことを特徴とする並列演算機システムの演算処理装置間の通信方法。

5

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は，通信手段および個別記憶装置を備えた多数の演算処理装置からなり，特に並列計算を目的とした並列計算機システム及びその演算処理装置の通信方法に関する。

【０００２】

【従来の技術】原子力施設をはじめとする大規模な施設の設計においては，例えば遮蔽設計などにおける放射線挙動計算，炉心設計における炉心性能予測解析などの大規模な計算がかなりの頻度で要求される。この要求に応えるためには大幅な計算速度の向上が必要である。このため最近では，通信手段と個別の記憶装置を備えた多数の演算処理装置を用いて，１台の演算処理装置しか持たない計算機を使用していたのでは得られないような高速度で，解析を行うことが考案されている。

【０００３】例えば炉心設計であれば，原子炉の炉心を複数の燃料集合体からなる幾つかのセグメントに分割し，それぞれのセグメントを１つの演算処理装置に対応させて，出力計算と熱水力計算を各々の演算処理装置で並列に計算させる。セグメント間での中性子束の流出入およびチャンネル間の冷却材の圧力バランスを解析する際には，前記通信手段によりセグメント境界の中性子束，各チャンネルの圧力損失のデータを演算処理装置間でやり取りすることで，空間的に連続した解析が行われる。

6

【０００４】また，遮蔽設計であれば，例えば原子炉の炉心，冷却材，遮蔽体などを含む全体系を幾つかの小領域に分割し，それぞれの小領域を１つの演算処理装置に対応させて，放射線束分布計算を各々の演算処理装置で並列に計算させる。小領域間での中性子束の流出入を解析する際には，前記通信手段により小領域境界の中性子束のデータを演算処理装置間でやり取りすることで，空間的に連続した解析が行われる。

【０００５】

【発明が解決しようとする課題】複数の演算処理装置を用いて並列に計算を行わせる際に，演算処理装置間の通信を行うことなく全く独立に計算を進めることができる例はまれであり，通常は演算処理装置間の通信を行いながら計算を進める。たとえば，４行４列の行列Ａ，Ｂの掛け算を４台の演算処理装置で実施して４行４列の行列Ｃを求める場合を考える。Ａ，Ｂ，Ｃの要素をそれぞれ a_{ij}，b_{ij}，c_{ij} で以下のように表記する。

【０００６】

【数６】

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix}$$

このとき，４台の演算処理装置のうちの１台においては例えば，

【０００７】

【数７】

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}$$

のように計算が行われる。

【０００８】この例から明らかなように，演算に使う側（a_{ij} または b_{ij}）については行或いは列全体についての要素のデータが必要である。また，演算の結果として得られる c_{ij} の方は，各々の演算処理装置に於いては部分的にしかデータが得られない。このことは，例えば次のステップで行列Ｃと行列Ａの掛け算を行う必要が生じたとき，計算で得られた要素だけではデータに不足が生じることを意味する。したがって，Ａ×Ｂ＝Ｃの計算を実施した後で残りの部分，上の式で言えば行列Ｃの少なくとも第１行と第２行のデータ及び第１列と第２列のデータは満たされた状態にしておかねばならない。

【０００９】これらの問題を一般化すると次のようになる。（ｎ×ｋ）個からなる配列Ｘ(nk)があり，これがｎ台の演算処理装置に分割され，例えば識別番号１の演算処理装置ではＸ(1)，Ｘ(2)，…Ｘ(k)，識別番号２の演算処理装置ではＸ(k+1)，Ｘ(k+2)，…，Ｘ(2k)　の計算結果を持っているものとする。この状態からｎ台の演算処理装置の間で通信を行うことにより，ｎ台の演算処理装置が配列Ｘ(nk)の計算結果を持っている状況を作る操作が必要となることがある。

【００１０】このときの通信は１対１であることが通信手段上の条件である。すなわち，例えば演算処理装置１から演算処理装置２にデータを転送する際には，演算処理装置２は演算処理装置１からデータを受けとる態勢になければならないのであって，このとき演算処理装置２が他の処理，例えば演算処理装置３にデータを転送しようとしたり演算処理装置４からデータを受けようとしたりすると，通信は失敗して計算は中断することとなる。通信が滞りなく行われるには送信側と受信側の混乱がないように通信の順序を予め決めておく必要がある。

【００１１】４台の演算処理装置を使う場合を例にとれば，容易に考えられる方法として次のものが挙げられ

る。以下，表記を簡略化するため演算処理装置１，２，３，４をそれぞれ＃１，＃２，＃３，＃４と書く。

（１）　送信－受信を１つずつ順次行う方法

[1] ＃１の計算結果→＃２，　[2] ＃１の計算結果→＃３，
[3] ＃１の計算結果→＃４，　[4] ＃２の計算結果→＃１，
[5] ＃２の計算結果→＃３，　[6] ＃２の計算結果→＃４，
[7] ＃３の計算結果→＃１，　[8] ＃３の計算結果→＃２，
[9] ＃３の計算結果→＃４，　[10] ＃４の計算結果→＃１，
[11] ＃４の計算結果→＃２，　[12] ＃４の計算結果→＃３

を順次実行する。

【００１２】ここで，[1]，[2]，[3]，…は処理のステップの番号を示す。演算処理装置をＮ台，１台に割り当てられたデータ量をｗとすれば，通信回数は

$2\times {}_{N}C_{2} = N(N-1)$

であり，データ移動量は

$2w\times {}_{N}C_{2} = wN(N-1)$

である。Ｎ＝４ならば通信回数は上述の12回である。この方法によれば，時間はかかるが通信上の混乱は避けられる。なお，${}_{p}C_{q}$ はｐ個の要素からｑ個の要素を選ぶ組合せの数を示す。

【００１３】（２）　代表の演算処理装置にデータを集めた後，各演算処理装置に配布する。

・データ集結

[1] ＃２の計算結果→＃１，　[2] ＃３の計算結果→＃１，
[3] ＃４の計算結果→＃１，…　を順次実行。
＃１に全データが揃う。

・全データ配布

[1] ＃１→＃２，　[2] ＃１→＃３，　[3] ＃１→＃４，…

を順次実行。　配列全体を＃２，＃３，＃４に送信する。

【００１４】この場合の通信回数は　２（Ｎ－１）回，データ移動量は集結時に（Ｎ－１）ｗ，配布時に　Ｎ（Ｎ－１）ｗである。この方法は（１）の方法に比べて通信回数は少ないが，全データ配布時に送信されるデータ量が多い点が短所である。

【００１５】また，通信の効率化を図った手法として次のものがある。

（３）　演算処理装置の１対１の組み合わせに対して並列・網羅的に通信を行う。　これは（１）の方法を改良したもので，例えば次のように行う。

[1] ＃１の計算結果→＃２，　＃３の計算結果→＃４　を同時に実行。
[2] ＃１の計算結果→＃３，　＃２の計算結果→＃４　を同時に実行。
[3] ＃１の計算結果→＃４，　＃２の計算結果→＃３　を同時に実行。
[4] ＃２の計算結果→＃１，　＃４の計算結果→＃３　を同時に実行。
[5] ＃３の計算結果→＃１，　＃４の計算結果→＃２　を同時に実行。
[6] ＃４の計算結果→＃１，　＃３の計算結果→＃２　を同時に実行。

【００１６】この通信方法によれば，通信が重複することも衝突することもなく，全データが４台の演算処理装置に行き渡る。演算処理装置がＮ台であれば通信回数は２（Ｎ－１），データ移動量は２（Ｎ－１）ｗである。Ｎ＝４であれば通信に要する時間は前述の（１）の方法の半分である。Ｎが大きくなるとともに差は広がる。

【００１７】（４）　演算処理装置の Binary treeにより代表の演算処理装置にデータを集めた後，各演算処理装置に配布する。これは（２）の方法を改良したもので，例えば次のように行う。

・データ集結

[1] ＃２の計算結果→＃１，＃４の計算結果→＃３を同時に実行。

[2] ＃３に集結された計算結果→＃１

・全データ配布

[3] ＃１→＃３

[4] ＃１→＃２，　＃３→＃４を同時に実行。

【００１８】この方法によれば，演算処理装置がＮ台であれば，通信回数は　$2\times \log_2 N$回，データ通信量は，集結時に（Ｎ－１）ｗ，配布時に　$Nw \log_2 N$である。Ｎ＝４であれば通信回数は（２）の方法の２／３，データ移動量は（２）の方法の　11／15である。Ｎが大きくなるとともに差は広がる。

【００１９】（３）の方法は（４）の方法に比べてデータ移動量は少ないが通信回数が多いため，扱う配列が小さい場合には適していない。（４）の方法は通信回数は少ないが，データ移動量が多いため，巨大な配列を扱う場合には適していない。

【００２０】よって，データ移動量と通信回数がともに最適化された，あらゆる条件に対して適用可能な一般化された手法が必要である。本発明は，このような点を考慮してなされたもので，通信によるデータの授受を並列に行えるようにすることで，演算処理装置間の通信回数およびデータの授受の際の待ち時間を最小限に抑えて高

速化を図ることができる並列計算機システム及びその演算処理装置間の通信方法を提供することを目的とする。

【００２１】

【課題を解決するための手段】上記目的を達成するため，本発明の請求項１記載の発明は，固有の識別子を有する少なくとも2^n台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，2^n個の小配列に分割して2^n台の演算処理装置に分配され各演算処理装置で演算処理されたデータ配列を再び１つの配列に集結する際に，2^n台の演算処理装置に識別番号０，１，…，2^n-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対しては，操作ｊの際に，識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することにより2^n台の演算処理装置間でｎ回の操作でデータ配列を集結させることを特徴とする。

【００２２】また，請求項２記載の発明は，固有の識別子を有する（2^n+k）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，（2^n+k）個の小配列に分割して（2^n+k）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，前記（2^n+k）台の演算処理装置に個別記憶手段及び通信手段を備えた（2^n-k）台の演算処理装置を加えた2^{n+1}台からなる演算処理装置群を形成し，この演算処理装置群を構成する2^{n+1}台の演算処理装置に識別番号０，１，…，$2^{n+1}-1$を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｍまで順次行い，ｊ＞０なるｊに対しては，操作ｊの際に，$N \leqq 2^n+k$なる識別番号Ｎの演算処理装置からはその演算処理装置の演算処理結果及び操作（ｊ－１）までで得られた演算処理結果を送信し，$N > 2^n+k$なる識別番号Ｎの演算処理装置からは操作（ｊ－１）までで得られた演算処理結果を送信することにより（2^n+k）台の演算処理装置において（ｍ＋１）回の操作でデータ配列を集結させることを特徴とする。

【００２３】また，請求項３記載の発明は，固有の識別子を有する（2^n+k）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，（2^n+k）個の小配列に分割して（2^n+k）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，この（2^n+k）個のデータ配列に（2^n-k）個の空の小配列を追加することで前記データ配列を小配列2^{n+1}個分の配列に拡張し，前記（2^n+k）台の演算処理装置に個別記憶手段及び通信手段を備えた（2^n-k）台の演算処理装置を加えた2^{n+1}台からなる演算処理装置群を形成し，この演算処理装置群を構成する2^{n+1}台の演算処理装置に識別番号０，１，…，$2^{n+1}-1$を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｍまで順次行い，ｊ＞０なるｊに対して，操作ｊの際に，識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することにより（2^n+k）台の演算処理装置において（ｍ＋１）回の操作でデータ配列を集結させることを特徴とする。

【００２４】また，請求項４記載の発明は，ｎ＞ｍなるｎ，ｍについて，固有の識別子を有する（2^n+2^m）台の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備え，この通信手段により各演算処理装置間でデータの授受を行う並列計算機システムにおいて，（2^n+2^m）個の小配列に分割して（2^n+2^m）台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，前記（2^n+2^m）台の演算処理装置を2^n台からなるグループG_1と2^m台からなるグループG_2に分割し，また前記データ配列を初めの2^n個の小配列からなる配列A_1とその後の2^m個の小配列からなる配列A_2の２つに分割し，この配列A_1，A_2をそれぞれグループG_1，G_2と対応づけて分配，演算処理を行い，グループG_1の2^n台の演算処理装置に識別番号０，１，…，2^n-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_1内でデータ配列を集結させる第１の工程と，グループG_2の2^m台の演算処理装置に識

11

別番号０，１，…，2^n-1を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_1内でデータ配列を集結させる第２の工程と，グループG_1からグループG_2の各演算処理装置に配列A_1を，グループG_2からグループG_1の各演算処理装置に配列A_2を送信する第３の工程とを有し，第１の工程と第２の工程を並列に実行した後に第３の工程を行なうことにより（2^n+2^n）台の演算処理装置においてデータ配列を集結させることを特徴とする。

【００２５】また，請求項５記載の発明は，固有の識別子を有する複数の演算処理装置と，これら各演算処理装置に各々対応する個別記憶装置および通信手段とを備えた並列計算機システムにおいて，

【数８】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

個の小配列（但し，$n_1>n_2>n_3>\cdots>n_k\geqq 0$）に分割して

【数９】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

台の演算処理装置に分配・演算処理されたデータ配列を再び１つの配列に集結する際に，これらの演算処理装置のうち

【数１０】

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

台をそれぞれグループG_1，G_2，…，G_kとしてｋ個のグループに分割するとともに，前記小配列のうち

【００２６】

【数１１】

$$2^{n_1},\ 2^{n_2},\ 2^{n_3},\ \cdots,\ 2^{n_k}$$

個の小配列をそれぞれ配列A_1，A_2，…，A_kとしてｋ個の配列に分割し，このｋ個の配列とｋ個のグループG_1，G_2，…，G_kとを１対１に対応づけて分配，演算処理を行い，$1\leqq p\leqq k$なる各ｐに対し，グループG_pの（2^{n_p}乗）台の演算処理装置に識別番号０，１，…を付与し，識別番号Ｎの演算処理装置に対し２進法で表した識別番号Ｎの2^iの位の数を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，前記データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置の間で相互に送受信する操作ｉをｉ＝０からｉ＝ｎ－１まで順次行い，ｊ＞０なるｊに対して，操作ｊの際に識別番号Ｎ，Ｎ’の演算処理装置間で各演算処理装置による演算処理結果に加えて操作（ｊ－１）までで得られた演算処理結果を送受信することによりグループG_p内の演算処理装置でデータ配列A_pを集結させるグループ内工程ｐを実行し，グループ内工程（ｋ－１）が終了した後，グループG_kの演算処理装置から配列A_kの演算結果をグループG_{k-1}の演算処理装置に送信するグループ間工程ｋを実行し，次に，グループG_pの各演算処理装置に集結された配列A_pの演算結果を，グループG_pの演算処理装置から$q>p$なる全てのｑに対しグループG_qに属する各演算処理装置に送信するとともに，グループG_qの演算処理装置から，グループG_p自身の演算結果である配列A_p及びグループG_{p+1}の演算処理装置から受信した配列A_{p-1}，…，A_kの演算結果をグループG_{p-1}の演算処理装置に送信するグループ間工程ｐを，ｐ＝ｋ－１からｐ＝２までｐに関して降順に実行することにより，

【００２７】

【数１２】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

台の演算処理装置においてデータ配列を集結させることを特徴とする。

【００２８】なお，この際には，ｋ個のグループ内工程１，２，…，ｋを並列に実行し，$1\leqq s\leqq k-1$なるｓに対して，グループ内工程ｓが終了した時点で順次グループ間工程（ｓ＋１）を実行することで、全体の通信に要する時間をさらに短縮することができる。

【００２９】また，請求項６記載の発明は，請求項４または５記載の並列計算機システムを用いて演算処理装置のグループ間でのデータ交換を行う場合，2^p台の演算処理装置からなるグループG_Aで集結され共有されているデータ配列Ａと，2^q台の演算処理装置（ｐ＞ｑ）からなるグループG_Bで集結され共有されているデータ配列Ｂとを，グループG_A，G_B間で相互に送受信する際に，グループG_Aのなかから選択される2^q台の演算処理装置をグループG_Bの各演算処理装置と１対１に対応させてグループG_Bの各演算処理装置にデータ配列Ａを送信する操作を並列に実施するとともに，グループG_Aを，それぞれが2^{p-q}台の演算処理装置からなる小グループα_1，α_2，…，α_r（$r=2^q$）に分割して，各々の小グループとグループG_Bのｒ個の各演算処理装置とを１対１に対応させ，小グループα_iのなかから選択される１台の演算処理装置に対して，小グループα_iに対応するグループG_Bの演算処理装置からデータ配列Ｂを送信した後，小グループα_iの演算処理装置間でデータ配列Ｂを送受信する操作ｉを，$1\leqq i\leqq r$なるｉに関して並列に実行するすることにより，2^p台の演算処理

12

装置と2^q台の演算処理装置にデータ配列Ａとデータ配列Ｂを共有させることを特徴とする。

【００３０】また，請求項７記載の発明は，請求項１乃至６のいずれか記載の並列計算機システムを用いて２台の演算処理装置間でデータを交換する工程は，演算処理装置の識別番号の大きい方から小さい方にデータを送る第１の送信工程と，演算処理装置の識別番号の小さい方から大きい方にデータを送る第２の送信工程とからなり，この第１の送信工程と第２の送信工程のうちから選択される１工程を先に行った後，続いて他の１工程を行うことを特徴とする。

【００３１】

【発明の実施の形態】本発明の実施の形態について，以下，図面を参照して説明する。図１は並列計算機システムの構成例を示すブロック図である。ここに示した並列計算機システムは，１台のホストの計算機１と８台の演算処理装置２－１，２－２，…，２－８で構成されている。ホストの計算機には記憶装置３と通信手段４，演算処理装置２－１，２－２，…，２－８の各々には，個別記憶装置５－１，５－２，…，５－８と通信手段６－１，６－２，…，６－８が備えられている。例えば，ホストの計算機で読み込んだ入力データ等は，通信手段４から通信手段６－１，６－２，…，６－８を通じて全演算処理装置に送信される。演算処理装置２－１，２－２，…，２－８では各々割り当てられた領域の計算を行い，必要に応じて演算処理装置間の通信によりデータの授受を行う。

【００３２】図１に示した並列計算機システムの構成に基き，本発明にかかる並列計算機システムの第１の実施の形態について説明する。図２は本実施の形態における並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【００３３】演算処理装置２－１，２－２，…，２－８の識別番号をそれぞれ０，１，…，７とし，これらを２進法の３桁の数として表示するとそれぞれ 000，001，010，011，100，101，110，111 となる。８×ｎ個のデータからなる配列Ａがｎ個のデータからなる８個の小配列a_1，a_2，…，a_8に分割されて，８台の演算処理装置２－１，２－２，…，２－８に割り当てられている。それぞれの演算処理装置で割り当てられた小配列のデータに関する演算処理を行った後，配列Ａの要素を全ての演算処理装置に於いて集めることを考える。なお，図２において各演算処理装置にかかれた０または１はそれぞれ分割された小配列を示しており，０は計算結果が未入力の状態を，１は計算結果が入力済みの状態を表す。

【００３４】第１ステップとして，2^0の位の数を反転（０ならば１，１ならば０とする）させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置０(000)は演算処理装置１(001)，演算処理装置３(011)は演算処理装置２(010)とｎ個のデータを交換する。各演算処理装置に２ｎ個の要素が集まる。

【００３５】第２ステップでは，2^1の位の数を反転させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置０(000)は演算処理装置２(010)，演算処理装置３(011)は演算処理装置１(001)とデータを交換する。この時，例えば演算処理装置０から演算処理装置２への送信では，演算処理装置０自身による演算結果の他に第１ステップで演算処理装置１から受信したデータを含む２ｎ個のデータを送信する。これにより各演算処理装置に４ｎ個の要素が集まる。

【００３６】最後に第３ステップとして，2^2の位の数を反転させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置０(000)は演算処理装置４(100)，演算処理装置３(011)は演算処理装置７(111)と４ｎ個のデータを交換する。各演算処理装置に８ｎ個の要素が集まり，操作が完了する。

【００３７】以上述べた通信方法は演算を2^3＝８個に分割した場合でありこの時のステップ数は３である。同様に，演算を2^4＝16個に分割し16台の演算処理装置において通信を行なう場合には，上述した８分割の場合に比べてさらに１ステップが必要となり，全部で４ステップとなる。一般に，演算をＮ個に分割しＮ台の演算処理装置において通信を行う場合は，上述の方法を流用して，ステップ数 $\log_2$ Ｎで通信が完了する。

【００３８】本実施の形態の作用効果について以下検証する。例えば配列の大きさをM(word)，演算処理装置の台数をＫとし，配列全体がＫ分割されて各演算処理装置に渡されているものとする。Ｋの値としては並列計算で最も一般的な条件である２のべき乗の場合，つまりＫ＝2^nと表される場合について考える。この状態から，演算処理装置間の通信によって演算処理装置全部が配列全体についてデータを把握している状況を作り出すのにかかる時間について考察する。一般にデータを送信するのに要する時間Ｔは

$$T = A + B \times W \quad \cdots\cdots (1)$$

と表せる。ここで，Ａは通信準備に要する時間で，送信するデータ量に関わらず１回の通信に必ず必要となる時間である。Ａの値はデータ量に依らない。Ｂ×Ｗはデータ量に比例する項であり，Ｗがデータ量（WORD数），Ｂが１word当たりの転送時間である。

【００３９】データの授受のステップ数は $\log_2$ Ｋ＝ｎである。各ステップで演算処理装置毎に送信と受信が１回づつ行われる。第ｍステップで授受されるデータ量は（Ｍ／Ｋ）×2^m [word] である。データ量M[word]のデータを全演算処理装置において集結させるのに必要な送受信の回数は各演算処理装置当り２ｎ回であり，送受信する総データ量は

【００４０】

15　　16

【数１３】

$$\sum_{m=1}^{n} \frac{M}{K} 2^m = \frac{M}{K} 2(2^n - 1) = 2M\left(1 - \frac{1}{K}\right) \text{ [word]}$$

となる。

【００４１】比較のため，従来法，例えばBinary treeの方式で１台の代表演算処理装置に全データを集めておき，同様にBinary treeの方式で全演算処理装置にデータを送信する場合の通信時間を次に求めてみる。全データを１台の演算処理装置に集めるのに要する送受信の回数は，代表演算処理装置においてｎ＝ $\log_2$ Ｋ回である。また，第ｍステップ（ｍ≦ｎ）で送信されるデータ量は（Ｍ／Ｋ）×2^{m-1} [word]である。よって，代表演算処理装置に全データを集めるのにかかる時間T_1は

$$T_1\ (K) = A \log_2 K + M(1 - 1/K) B \quad \cdots\cdots\cdots\cdots (3)$$

となる。

【００４２】代表演算処理装置から各演算処理装置にデータを配布する際のステップ数は$\log_2$ Ｋで，演算処理装置あたり通信回数も最大で $\log_2$ Ｋ回である。ただし，各ステップ毎にM[word]のデータが送信される。よって，各演算処理装置にデータを配布する際にかかる時間T_2は

$$T_2\ (K) = A \log_2 K + (M \log_2 K) B \quad \cdots\cdots\cdots\cdots\cdots (4)$$

となる。したがって，全通信時間$T_0 = T_1 + T_2$は

$$T_0\ (K) = 2A \log_2 K + M(1 - 1/K + \log_2) B \quad \cdots\cdots (5)$$

となる。

【００４３】図３及び図４のグラフは，横軸に演算処理装置台数，縦軸に通信に要する時間をとって，演算処理装置台数増加に伴う通信時間の増加の関係を示しており，従来のBinary Tree の通信方式による (5)式の関係と，本実施の形態により通信を効率化した (2)式の関係を，比較して示している。このグラフ中の曲線のうち実線で示した符号10ａ，10ｂが本実施の形態の (2)式の場合，破線で示した符号11ａ，11ｂが従来の (5)式の場合を示している。図３に示した符号10ａ，11ａを付した曲線は，通信されるデータ量が少なく，(1) 式のＡ（通信立ち上げ時間）が全通信時間Ｔのほぼ半分を占める状況を，また図４に示した符号10ｂ，11ｂを付した曲線は，通信されるデータ量が多く，(1) 式のＡ（通信立ち上げ時間）が全通信時間Ｔに比べて十分小さい状況を想定している。このグラフからも明らかなように，本実施の形態によれば，演算処理装置の台数が少数の場合，多数の場合何れも従来の方法より通信に要する時間を少なくすることができる。すなわち，本実施の形態により，データの授受の際の待ち時間を最小限に抑え，計算の高速化を図ることができる。

【００４４】なお，例えば演算処理装置の台数が16台からなる並列計算機システムにおいて，その内の８台の演算処理装置の間で上述の３ステップからなる配列の分割分配，演算集結を行うなど，複数の演算処理装置のうち２の冪乗の台数だけ抜き出してこれらに通信制御用の識別番号を付与し，この台数に適応して上述した方法で配列の分割分配，演算処理を行なうものとしてもよい。

【００４５】上記第１の実施の形態においては，関係する演算処理装置の台数が２の冪乗であることを前提としている。一般的な条件として演算処理装置の台数が２の冪乗でない場合，すなわち台数が $2^n + k$ 等として表である。よって，本発明を適用した場合の全通信時間Ｔは，

$$T\ (K) = 2A \log_2 K + 2M(1 - 1/K) B \quad \cdots\cdots\cdots\cdots (2)$$

となる。

【００４６】本発明にかかる並列計算機システムの第２の実施の形態について説明する。ここでは，例えば並列計算の配列を６分割して，６台の演算処理装置（識別番号を０，１，…，５とする。）に割り当てる場合について説明する。図５は本実施の形態における並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。この際のデータ処理には，前記６台の演算処理装置のほかに２台の演算処理装置（識別番号を６，７とする。）を用いることとする。

【００４７】第１ステップとして，2^0 の位の数を反転させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置０(000) は演算処理装置１(001) と，演算処理装置３(011) は演算処理装置２(010) と，それぞれｎ個のデータを交換する。演算処理装置６(110) と演算処理装置７(111) は交換すべきデータがないので休止する。この時点で，演算処理装置０～５に２ｎ個のデータが集められる。

【００４８】第２ステップでは，2^1 の位の数を反転させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置４(100) は演算処理装置６(110) とのデータ交換となるが，この時点で演算処理装置６(110) は送信すべきデータがないので，演算処理装置４からデータを受信するのみとする。このデータ交換により，演算処理装置０～３に４ｎ個のデータが，演算処理装置４～７には２ｎ個のデータが集められる。

【００４９】第３ステップでは，2^2 の位の数を反転させた数を識別番号としてもつ演算処理装置との間でデータを交換する。例えば演算処理装置６(110) は演算処理装置２(010) との交換である。演算処理装置６から演算

処理装置２へは２ｎ個のデータ，演算処理装置２から演算処理装置６へは４ｎ個のデータを送信する。このようにして６ｎ個のデータが８台の演算処理装置全てに行き渡る。

【００５０】本実施の形態においては，一般に（2^n ＋ｋ）台の演算処理装置に対して，（2^n －ｋ）台の演算処理装置を加えた2^{n+1} 台の演算処理装置群を構成し，この演算処理装置群に対して上述の第１の実施の形態で詳述したステップにより並列計算を行うものとする。これにより，２の冪乗ではない台数の演算処理装置に対しても２の冪乗の場合に準じた構成とすることで，上記第１の実施形態と同様の作用効果を得ることができる。

【００５１】次に本発明にかかる並列計算機システムの第３の実施の形態を説明する。本実施の形態における演算処理装置間の通信方法について，例として，配列を６個の小配列分割して６台の演算処理装置（識別番号０，１，…，５）に割り当てている場合について説明する。まず前記配列を小配列２個分拡張し，拡張した部分には０を埋める。例えば12個の要素からなる配列（3,1,4,1,5,9,2,6,5,3,5,8)であれば，４個の要素からなる配列(0,0,0,0）を追加して，16の要素からなる配列（3,1,4,1,5,9,2,6,5,3,5,8,0,0,0,0)とする。演算処理装置としては前記６台の演算処理装置のほかに２台の演算処理装置（識別番号６，７とする）を加えた８台の演算処理装置を用いる。この後は，上記第１の実施の形態において詳述した手順により，８台の演算処理装置間で通信を行いデータを交換する。

【００５２】本実施の形態においては，一般に（2^n ＋ｋ）台の演算処理装置に対して，（2^n －ｋ）台の演算処理装置を加えた2^{n+1} 台の演算処理装置群を構成し，また配列についてもその要素を2^{n+1} 個に拡張して各演算処理装置に分配し，上記第１の実施の形態と同様の方法で並列計算及びデータの集結を行うものとする。これにより，２の冪乗ではない台数の演算処理装置に対しても２の冪乗の場合に準じた構成とすることで，上記第１の実施形態と同様の作用効果を得ることができる。

【００５３】次に，本発明にかかる並列計算機システムの第４の実施の形態について説明する。第２及び第３の実施の形態は，配列の分割数が２の冪でない場合，すなわち（2^n ＋ｋ）個に分割される場合について，2^{n+1} 台の演算処理装置によってデータ配列を１個に集結する方法について述べたものである。これに対し本実施の形態は，配列の分割数が，2^n ＋2^m （ｎ＞ｍ）である場合に対し，（2^n ＋2^m ）台の演算処理装置で処理するものである。

【００５４】本実施の形態における並列計算機システムの演算処理装置間の通信方法として，ここではまず例として，配列を６分割して６台の演算処理装置（識別番号０，１，…，５）に割り当てている場合について説明する。図６はこの場合の演算処理装置間通信方法を時系列で示すチャートである。

【００５５】まず，６台の演算処理装置を２つのグループに分割する。演算処理装置グループ１は識別番号０～３の４台で構成される。演算処理装置グループ２は識別番号４～５の２台で構成される。次に，演算処理装置グループ１の４台間，および演算処理装置グループ２の２台間で，上述の第２の実施の形態における手順により，各々のグループでデータを集結させる。図６における第１及び第２ステップがこれに相当する。

【００５６】この後，グループ１とグループ２でデータ交換を次の手順で行う。

[1] 演算処理装置４→演算処理装置０，演算処理装置５→演算処理装置２を同時並列に実施。（図６の第３ステップに相当。）

[2] 演算処理装置１→演算処理装置４，演算処理装置３→演算処理装置５を同時並列に実施。（図６の第４ステップに相当。）

[3] 演算処理装置０→演算処理装置１，演算処理装置２→演算処理装置３を同時並列に実施。（図６の第５ステップに相当。）

この方法により，６台の演算処理装置によってデータ配列を集結させることができる。

【００５７】また，本実施の形態のもう一つの例として，配列を10分割して10台の演算処理装置（識別番号０，１，．．．９）に割り当てている場合について説明する。図７はこの場合における演算処理装置間通信方法を時系列で示すチャートである。

【００５８】まず，10台の演算処理装置を２つのグループに分割する。演算処理装置グループ１は識別番号０，１，…，７の８台で構成される。演算処理装置グループ２は識別番号８，９の２台で構成される。次に演算処理装置グループ１の８台の演算処理装置間，および演算処理装置グループ２の２台の演算処理装置間で，上記第２の実施の形態において述べた方法により，各々のグループでデータを集結させる。図７における第１，第２及び第３ステップがこれに相当する。

【００５９】この後，グループ１とグループ２でデータ交換を次の手順で行う。

[1'] 演算処理装置８→演算処理装置０，演算処理装置９→演算処理装置４を同時並列に実施。（図７の第４ステップに相当。）

[2'] 演算処理装置３→演算処理装置８，演算処理装置７→演算処理装置９，演算処理装置０→演算処理装置２，演算処理装置４→演算処理装置６を同時並列に実施。（図７の第５ステップに相当。）

[3'] 演算処理装置０→演算処理装置１，演算処理装置４→演算処理装置５，演算処理装置２→演算処理装置３，演算処理装置６→演算処理装置７を同時並列に実施。（図７の第６ステップに相当。）

【００６０】この方法により，６台の演算処理装置によ

19

ってデータ配列を集結させることができる。なお，グループ２からグループ１に送信されたデータのグループ２内の分配は Binary Treeの方式によっている。

【００６１】以下，本発明にかかる並列計算機システムの第５の実施の形態について説明する。本実施の形態における演算処理装置間の通信方法は，上記第５の実施の形態の通信方法を一般化したものである。以下，例として配列を22分割して22台の演算処理装置（識別番号０，１，…，21）に割り当てている場合について説明する。図８及び図９はこの配列22分割の場合における演算処理装置間通信方法を時系列で示すチャートである。図８において第１ステップから第４ステップまでを，図９において第５ステップから第８ステップまでを示した。

【００６２】$22=2^4+2^2+2^1$ であるから，まず，演算処理装置を次の３グループに分ける。

グループ１；　識別番号０，１，…，15の演算処理装置（16台）

グループ２；　識別番号16，17，18，19の演算処理装置（４台）

グループ３；　識別番号20，21の演算処理装置（２台）

【００６３】次に，演算処理装置グループ１の16台間，演算処理装置グループ２の４台間，および演算処理装置グループ３の２台間で，上記第１の実施の形態の方法により各々のグループでデータを集結させる。これは図８に示した第１ステップから第４ステップまでが相当する。

【００６４】この後は，上記第２或いは第３の実施の形態において説明した方法と同様の手順により，データの

グループ１の小グループ１；　演算処理装置０，１，２，３

小グループ２；　演算処理装置４，５，６，７

小グループ３；　演算処理装置８，９，10，11

小グループ４；　演算処理装置12，13，14，15

とする。

【００６８】この各小グループから１台ずつ演算処理装置を選択する。ここでは演算処理装置０，４，８，12を選択する。この４台の演算処理装置に対して，それぞれグループ２の演算処理装置16，17，18，19から，グループ２及びグループ３に関して集結されたデータを送信する。これは図９に示した第６ステップに相当する。

【００６９】次に，グループ１の各小グループにおいて，従来のBinary Tree の方式で演算処理装置間でグループ２，３に関するデータの送受信を行ない，小グループの全演算総理装置においてグループ１，２，３のデータを集結させる。例えば小グループ１においては演算処理装置０から演算処理装置２に対してデータを送信し，次に演算処理装置０，２からそれぞれ演算処理装置１，３に対してデータの送信を行う。他の小グループにおいても同様である。これは図９に示した第７ステップ及び第８ステップに相当する。こうして，全ての22台の演算処理装置において22個のデータ配列の集結を完了する。

20

グループ間交換を行う。以下そのデータの通信方法を順を追って説明する。まず，第２ステップでグループ２においてデータの集結が終了するが，その時点で既にグループ３のデータの集結は完了しているから，次のステップとして，グループ２の演算処理装置16，18とグループ３の演算処理装置19，20との間でそれぞれデータの交換が行なわれる。これは図８に示したグループ２とグループ３における第３ステップに相当する。この時点で，グループ３の全ての演算処理装置にはグループ２及びグループ３におけるデータがすべて格納された状態となる。

【００６５】次に，グループ２及びグループ３の全てのデータが格納されたグループ２の演算処理装置16，18から，それぞれグループ２の演算処理装置17，19に対してグループ３より受信したデータが送信される。これは図８に示したグループ２における第４ステップに相当する。

【００６６】グループ１においては第４ステップで各演算処理装置間でデータの集結が終了するが，次のステップとして，グループ１と，グループ２，３との間でデータの送受信を行う。まず，グループ１の演算処理装置０，１，２，３，４，５から，それぞれグループ２，３の演算処理装置16，17，18，19，20，21に対してデータが送信される。これによりグループ２，３においてはグループ１，２，３の22台の全ての演算処理装置のデータの集結が完了する。これは図９に示した第５ステップに相当する。

【００６７】次に，グループ１の16台の演算処理装置を４つの小グループに分割する。すなわち，

【００７０】一般に，２の冪乗では表されない台数の演算処理装置におけるデータ配列は，以上説明した方法によって集結させることができる。まず，ｋ個の整数

$n_1, n_2, n_3, \cdots, n_k$　（但し，$n_1 > n_2 > n_3 > \cdots > n_k \geq 0$）

を用いて，並列計算機システムの演算処理装置の台数を

【００７１】

【数１４】

$$2^{n_1}+2^{n_2}+2^{n_3}+\cdots+2^{n_k}$$

と表す。また，データ配列をこの台数と同数の小配列に分割し，各演算処理装置に分割して演算処理を行なうものとする。並列計算機システムの演算処理装置のうち，

【００７２】

【数１５】

$$2^{n_1}, 2^{n_2}, 2^{n_3}, \cdots, 2^{n_k}$$

台をそれぞれグループG_1，G_2，…，G_kとして，並列計算機システムの演算処理装置をｋ個のグループに分

21

割する。同様にデータ配列の小配列の

【００７３】

【数１６】

$$2^{n_1}, 2^{n_2}, 2^{n_3}, \cdots, 2^{n_k}$$

個をそれぞれ配列A_1，A_2，…，A_kとしてｋ個の配列に分割する。

【００７４】次に，$1 \leqq p \leqq k$なるすべてのｐに対して，以下の『』内に定義する操作（以下，グループ内工程ｐという。）を行う。但し，グループ内工程１，…，ｋは並列して行うこととする。

【００７５】『 グループG_pの（２のn_p乗）個の演算処理装置に識別番号０，１，…，（２のn_p乗－１）を付与する。次に，$0 \leqq q \leqq p-1$なるｑに対し，以下の《 》内に定義する操作ｑを，ｑ＝０からｑ＝ｐ－１まで順次行なう。

《 識別番号Ｎの演算処理装置に対し，２進法で表した識別番号Ｎの2^qの位を反転させた番号Ｎ’を識別番号とする演算処理装置を対応させ，データ配列の演算処理結果を識別番号Ｎの演算処理装置と識別番号Ｎ’の演算処理装置との間で相互に送受信する。但し，ｑ＞０なるｑに対しては，操作ｑの際に，識別番号Ｎ，Ｎ’の演算処理装置間で，各演算処理装置による演算処理結果に加えて操作（ｑ－１）までで得られた演算処理結果を合わせて送受信することとする。 》

この操作により，グループG_pの（２のn_p乗）台の演算処理装置で，データ配列の集結を行う。 』

グループの設定方法により，グループ内工程１，…，ｋを並列に行なったとき，グループ内工程ｋが最初に終了し，以下，グループ内工程（ｋ－１），…，２，１の順に終了する。このことを考慮して，以下の {{ }} に定義する操作（以下，グループ間工程ｐという。）を，ｐ＝ｋ－１からｐ＝１までｐに関して降順に行うこととする。

【００７６】{{ グループ内工程ｐが終了した後，グループG_pの各演算処理装置に集結された配列A_pのデータを，グループG_pの演算処理装置から，グループG_{p+1}，…，G_kに属する全ての演算処理装置に送信する。すなわち，グループG_pに属する（２のn_p乗）台の演算処理装置のうち

【００７７】

【数１７】

$$2^{n_{p+1}} + \cdots + 2^{n_k}$$

台を選択して，これら選択された演算処理装置とグループG_{p+1}，…，G_kに属する演算処理装置とを１対１に対応させ，グループG_pからグループG_{p+1}，…，G_kへの配列A_pのデータ送信を行う。次に，グループG_{p+1}からグループG_pへのデータの送信を行う。（２のn_p乗）台の演算処理装置からなるグループG_pを，それぞれが

22

【００７８】

【数１８】

$$2^{n_p - n_{p+1}}$$

台の演算処理装置からなる小グループα_1，…，α_rに分割する。この小グループの数ｒは，

【００７９】

【数１９】

$$r = 2^{n_{p+1}}$$

である。ここで，グループG_{p+1}に属する演算処理装置をb_1，…，b_rと表記する。 グループG_pの小グループα_1，…，α_rと，グループG_{p+1}に属する演算処理装置をb_1，…，b_rとを１対１に対応させて，グループG_{p+1}の演算処理装置b_iから対応する小グループα_iのうちから選択された１台の演算処理装置a_iに，グループG_{p+1}において集結された配列A_{p+1}のデータを送信する操作を，$1 \leqq i \leqq r$なる全てのｉについて並列に行う。このとき，$p < k-1$の場合，演算処理装置b_iからa_iへは，グループG_{p+2}，…，G_kより受信したデータ配列A_{p+2}，…，A_kを含めて送信するものとする。

【００８０】この後，各小グループα_iにおいて，演算処理装置a_iからa_i以外の全ての演算処理装置に対して，従来のBinary Tree の方式でデータの送信を行なう。これにより，グループGp の全ての演算処理装置に対してデータ配列A_p，…，A_kに関するデータ配列の集結が完了する。 }}

この方法により，一般に複数台の演算処理装置によって各演算処理装置において分散され並列計算されたデータ配列を，効率よく集結させることができるから，計算の高速化を図ることができる。

【００８１】

【発明の効果】以上説明したように本発明によれば，並列計算機システムの演算処理装置間の通信方法の効率をより向上させることにより，データの授受の際の待ち時間を最小限に抑えることができるから，並列計算機システムにおいて実施される大規模な計算の高速化を図ることができる。

【図面の簡単な説明】

【図１】本発明の第１の実施の形態における並列計算機システムの構成を示すブロック図である。

【図２】本発明の第１の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【図３】通信されるデータ量が少ない場合の本発明の第１の実施形態及び従来の通信方法を用いた場合の演算処理台数と通信時間の相関を示すグラフである。

【図４】通信されるデータ量が多い場合の本発明の第１

の実施形態及び従来の通信方法を用いた場合の演算処理台数と通信時間の相関を示すグラフである。

【図５】本発明の第２の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【図６】本発明の第２の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【図７】本発明の第４の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【図８】本発明の第５の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

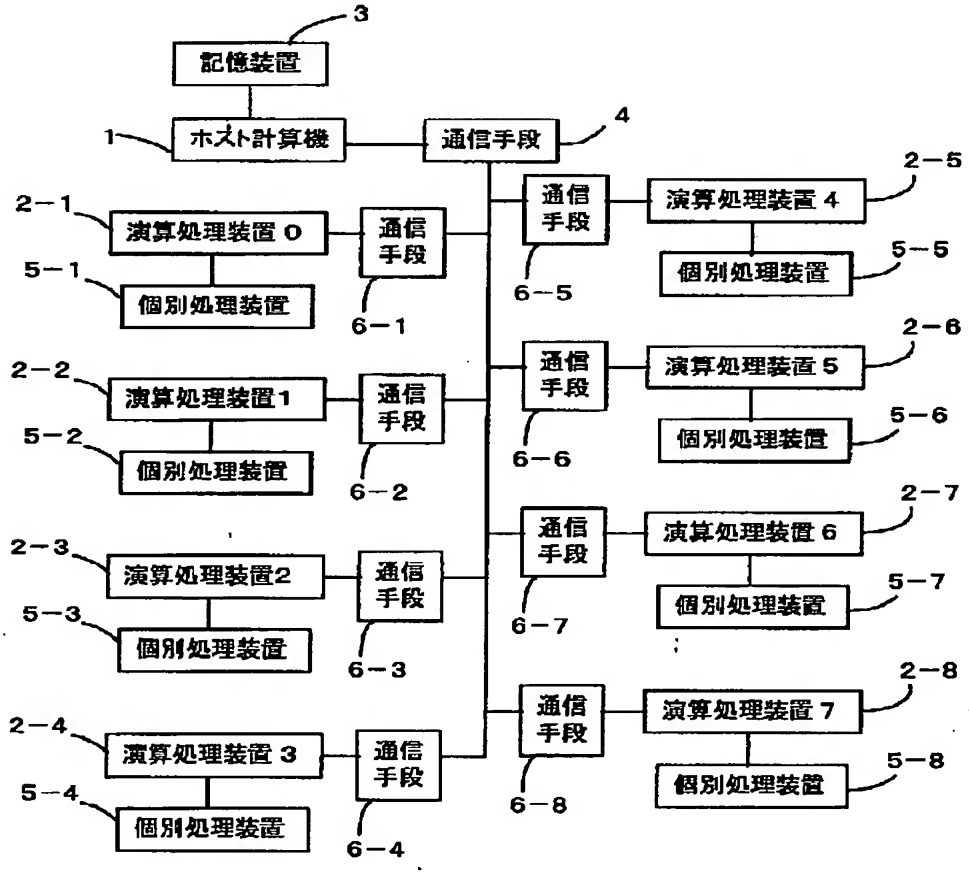
【図９】本発明の第５の実施の形態にかかる並列計算機システムの演算処理装置間の通信方法を時系列で示すチャートである。

【符号の説明】

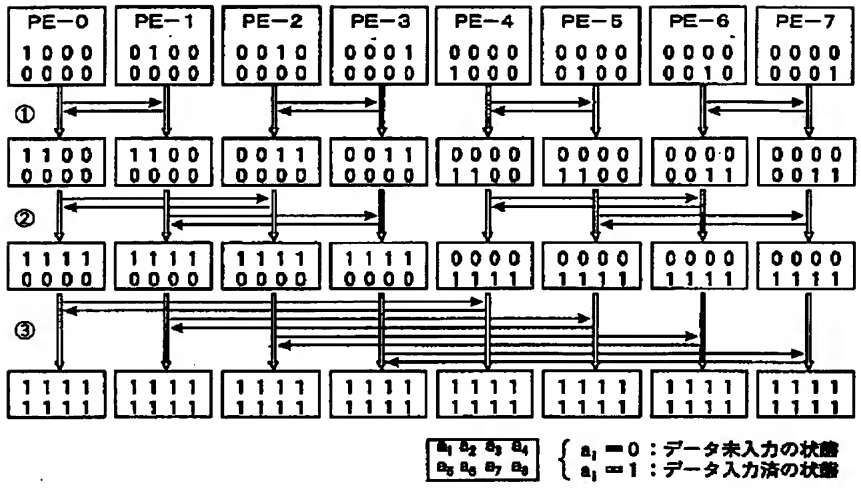
１…ホスト計算機，　２－１…演算処理装置，　３…記憶装置，４…通信手段，　５－１…個別処理装置，　６－１…通信手段

10　10ａ，10ｂ…本発明の第１の実施の形態における演算処理装置の台数と通信に要する時間の関係を示す曲線

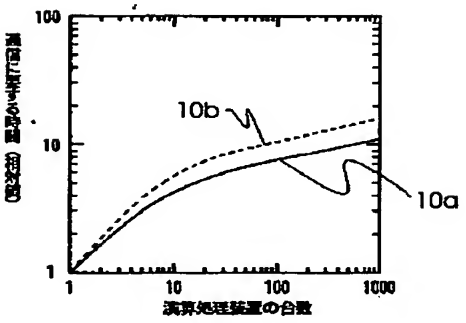
【図１】



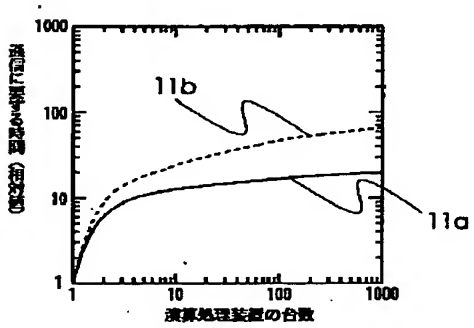
【図２】


PE−0
PE−1
PE−2
PE−3
PE−4
PE−5
PE−6
PE−7
①
②
③
a_1 a_2 a_3 a_4
a_5 a_6 a_7 a_8
a_i = 0 ：データ未入力の状態
a_i = 1 ：データ入力済の状態

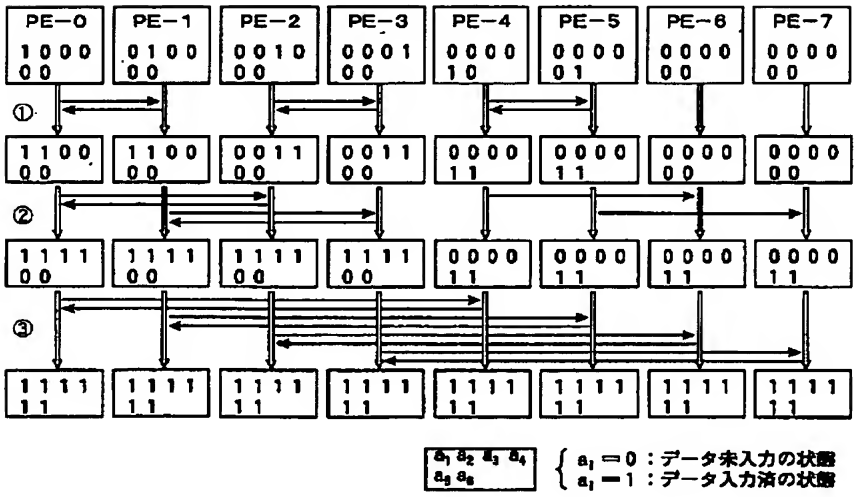
【図３】


100
10
1
10b
10a
1
10
100
1000
演算処理装置の台数

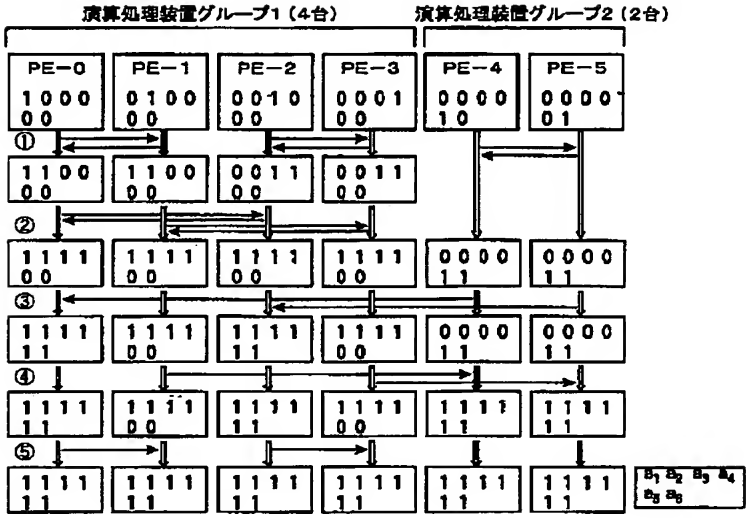
【図４】


1000
100
10
1
11b
11a
1
10
100
1000
演算処理装置の台数

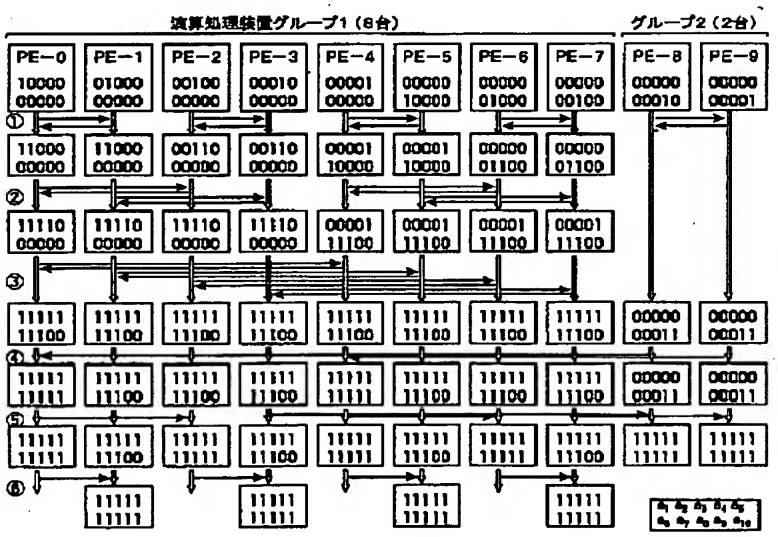
【図５】


PE−0
PE−1
PE−2
PE−3
PE−4
PE−5
PE−6
PE−7
①
②
③
a_1 a_2 a_3 a_4
a_5 a_6
a_i = 0 ：データ未入力の状態
a_i = 1 ：データ入力済の状態

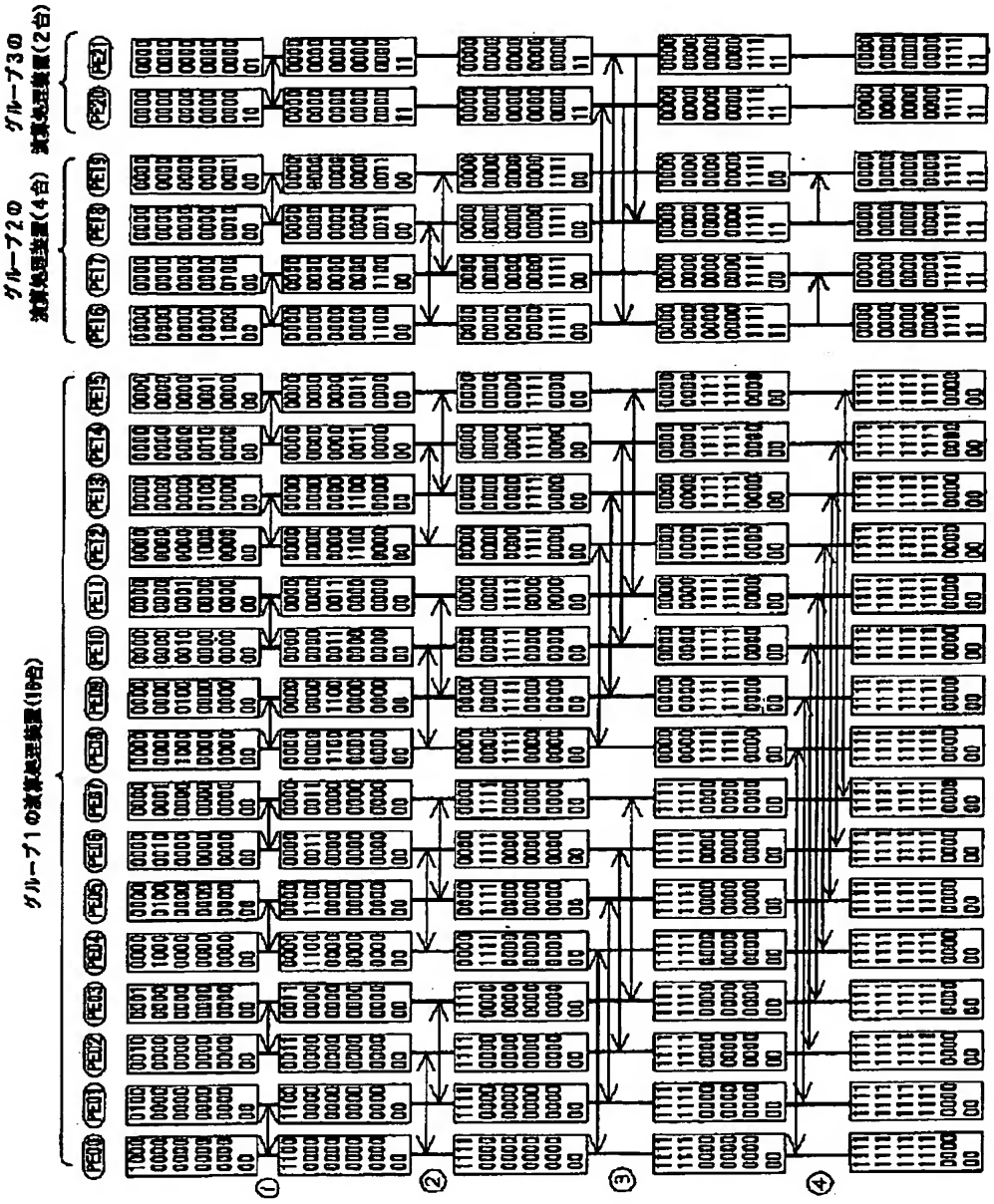
【図６】


演算処理装置グループ1（4台）
演算処理装置グループ2（2台）
PE−0
PE−1
PE−2
PE−3
PE−4
PE−5
①
②
③
④
⑤

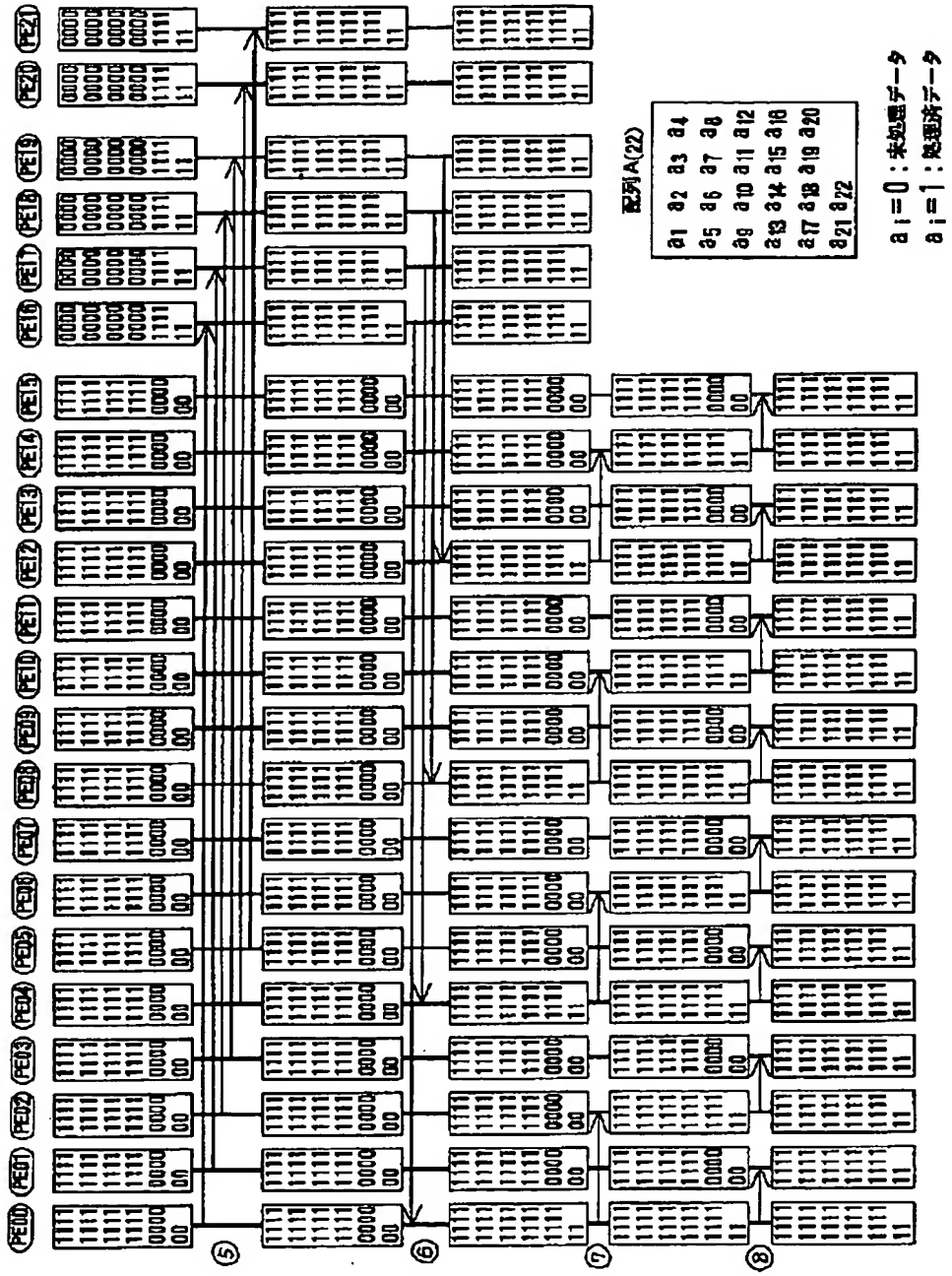
【図７】


演算処理装置グループ1（8台）
グループ2（2台）
PE−0
PE−1
PE−2
PE−3
PE−4
PE−5
PE−6
PE−7
PE−8
PE−9
①
②
③
④
⑤
⑥

【図８】


グループ3の
演算処理装置(2台)
グループ2の
演算処理装置(4台)
グループ1の演算処理装置(10台)
PE21
PE20
PE19
PE18
PE17
PE16
PE15
PE14
PE13
PE12
PE11
PE10
PE9
PE8
PE7
PE6
PE5
PE4
PE3
PE2
PE1
PE0
①
②
③
④

【図９】


PE21
PE20
PE19
PE18
PE17
PE16
PE15
PE14
PE13
PE12
PE11
PE10
PE09
PE08
PE07
PE06
PE05
PE04
PE03
PE02
PE01
PE00
配列A(22)
a1 a2 a3 a4
a5 a6 a7 a8
a9 a10 a11 a12
a13 a14 a15 a16
a17 a18 a19 a20
a21 a22
ai=0：未処理データ
ai=1：処理済データ
⑤
⑥
⑦
⑧